

## 0.1 oefeningen

(oefeningen, voorbeelden fouten softwareprojecten.)

- De programmeur/tester, ze hielden waarschijnlijk enkel de laatste 2 cijfers van de leeftijd bij. Dus 104 was dan 04 in de database.
- Programmeur/tester, want zij hielden geen rekening met schrikkeljaren.
- De tester/analyst/ontwerp. Zij hielden geen rekening met de startprocedure en veiligheid.
- De projectleider/klant was in de fout, want hij had geen rekening gehouden met de requirements van de klant omtrent de levertijd en de kost. De klant zelf veranderde waarschijnlijk te veel van gedachten en veranderden teveel requirements.
- De analyst/programmeur/tester/iedereen. er was niet genoeg info over wie in de fout was.
- De programmeur en de ontwerper waren in de fout, want zij hebben het zo onnodig complex gemaakt, en de programmeur heeft niet 'geprotesteerd'.

## 1 Behoeftenanalyse

vereisten is het modelleren van verhalen. DENK SIMPEL, GRUGBRAIN. Wat wil de klant, wat moeten wij programmeren. Werken volgens de requirements.

Requirements kunnen worden onderverdeeld. Functioneel vs niet functioneel. Functionele requirements in een berichtensysteem zijn dingen zoals inloggen of teksten schrijven, de teksten versturen. Beschrijft de inputs van het programma op de outputs, nbeschrijft dus wat het systeem moet kunnen.

De niet-functionele requirements, het moet passen binnen de layout van de firma

Use cases: definitie. Een use case omvat alle maniere waarop het systeem gebruikt kan worden om een bepaald doel te behalen.

Het verteld niet HOE het systeem moet worden geïmplementeerd.

Use case diagram, Een diagram waarin elke use case staat, en wie de (primaire) actoren zijn van die use case. Bv. Wie registreerd, wie speelt een spel?. Een gewone gebruiker kan enkel registreren. En daarna wordt het een speler, en een speler moet kunnen aanmelden, spelen etc.../

### 1.1 Use case diagram, bibliotheek.

3 rollen hebben, en minstens 1 rol per activiteit.

- Lener (klant) aanmelden, boeken uitlenen, boeken terugbrengen.

- Bibliothecaris (beheerder) boeken toevoegen, boeken verwijderen, boeken zoeken, boeken uitlenen, boeken terugbrengen. Gebruikers toevoegen. Aanmelden
- niet geregistreerde gebruiker (gebruiker) boeken zoeken, registreren.

Een include is het feit dat bv usecase A **ALTIJD** zal moeten gebeuren met usecase B. Om geld te storten moet je **ALTIJD** eerst aanmelden met je pincode. Het is dus een include, want het is **VERPLICHT**. Extends; use case A voert use case B uittijdenseenalternatiefverloop. Die use kan kan soms een andere use case bevatten.

## 1.2 Use case diagram, bibliotheek.

- Nieuw lid registreren
- Lid Aanmelden, boek uitlenen
- Werknemer aanmelden, Iemand registreren, Nieuw boek toevoegen, uitleningen controlleren
- werkgever Iemand registreren, boek toevoegen, aanmelden, boek uitlenen.
- Een extend is dus een alternatief verloop, een include is een verplichte use case.
- Een include is dus een verplichte use case.
- Om een boek uit te lenen moet je geregistreerd zijn/aangemeld zijn, dus is dus een include use case
- Een exclude voorbeeld, Je kan geen boeken uitlenen als je nog boetes hebt. Domeinregel dus.